논문 2009-46CI-3-5

반수동형 센서 태그 데이터의 효율적인 읽기 기법

(Techniques for Efficient Reading of Semi-Passive Sensor Tag Data)

김 수 한*, 류 우 석*, 홍 봉 희**

(Soohan Kim, Wooseok Ryu, and Bonghee Hong)

요 약

본 논문은 반수동형 센서 태그의 센서 데이터를 효율적으로 읽기에 대한 주제를 고찰한다. 콜드 체인 관리 시스템에서는 데이터의 손실이 없이 센서 데이터가 완전하게 읽히고 빠른 처리 시간 안에 센서 태그 데이터를 읽는 것을 요구한다. 하지만, 센서 태그 데이터를 읽을 때 전파방해나 장애물과 같은 RF 환경에 의한 간섭을 받게 된다. 본 연구에서는 이러한 원인이 센서 데이터의 손실을 발생시키고 데이터 손실이 발생하였을 때 센서 데이터를 읽는데 더 많은 시간이 걸리는 것을 알아냈다. 이러한 문제점을 해결하기 위하여, 우리는 센서 데이터의 효율적인 읽기를 보장해주는 트랜잭션 처리 메커니즘을 제안한다. 이를 위해, 우리는 읽기 트랜잭션 실행을 위해 동적 패킷 크기 기법과 데이터 복구 기법을 설명한다. 이러한 기법은 대용량의 데이터를 위한 처리시간 단축뿐만 아니라 읽기 연산의 신뢰성 또한 개선시킨다. 본 논문에서 데이터의 손실도 없고 많은 시간 소요도 없는 센서 데이터의 효율적인 읽기 수행의 개선에 기여함을 확인 하였다.

Abstract

This paper investigates the issue of efficient reading for sensor data of semi-passive sensor tag. The Cold Chain management system requires complete sensor data without data loss and the short processing time of reading sensor tag data. However, reading the sensed data could be interfered by RF environment such as a jamming, obstacle and so on. This study found that it could lead to loss of the sensed data and takes much time to read it when data loss is occurred. To solve this problem, we propose the transaction processing mechanism that guarantees efficient reading of the sensed data. To do this, we present the technique of dynamic packet size and technique of data recovery to execute read transaction. These techniques improve the reliability of reading operation as well as speed up of read process for the large capacity data. This paper contributes to the improvement of efficient reading of sensed data without any loss of data and large time required.

Keywords : transaction processing, efficient reading, dynamic packet size, semi-passive sensor tag, RFID middleware

Ⅰ. Introduction

Radio frequency identification (RFID) technology is a wireless communication system using radio-frequency waves to transfer data between readers and movable tagged objects. RFID system does not require the transponder to be in line of sight and it allows multiple tag identification. There are several kinds of tag. One of the tags which are semi-passive tag contains sensor and it can record sensor value at its own memory[1].

A lot of logistics environments such as Cold Chain use sensor tags for product management. Semi-passive sensor tag could be used to manage attributes of products such as temperature of

* 학생회원, ** 정회원, 부산대학교 컴퓨터공학과
(Department of Computer Engineering, Pusan National University)

※ This work was supported by the grant No. B0009720 from the Regional Technology Innovation Program of the Ministry of Knowledge Economy(MKE).

접수일자: 2009년3월27일, 수정완료일: 2009년5월4일

perishable goods (e.g. fish, meat). It is useful device for improving the control during the transport chain and detecting weaknesses by identifying specific problem areas where corrective actions are necessary[2]. So semi-passive sensor tag is used in the environment where the logging plays a very important role sensing information.

When reader read sensed data from semi-passive sensor tag, data loss may be occurred because of the capacity of tag data is increased and more time will be needed to read it. We do the case study about the situation that data loss occurs. Noisy environment and movement of tag are the two main causes. For these reasons it cannot perfectly manage the perishable products in logistics environment likes Cold Chain.

To solve the problem of data loss, we propose transaction processing methods for guaranteeing complete reading and high speed of read processing while it read sensed data from semi-passive sensor tag. Transaction processing uses Dynamic Packet Size technique and incremental read method. Dynamic Packet Size technique provides considerable reduction of the time required when reading a sensed data in noisy environment. And incremental read method is to read sensed data one by one according to the packet size. Using these two methods properly it guarantees efficient reading of transaction processing.

We make a performance evaluation about proposed methods, and confirm that it provides efficient reading of the sensed data. In the performance evaluation, we compare existing read processing method with proposed transaction processing method. We found the result that our proposed transaction processing method provides complete result data at the short processing time.

This paper is organized as follows. Section II presents target environment. In Section III, presents problems of inefficient read of the sensed data. Transaction processing for efficient reading is presented in Section IV. In Section V, we present performance test over proposed transaction processing. We analyze functions of RFID system in Section VI. A summary is presented in Section VII.

II. Target Environment

The Cold Chain management system manages status of products using sensor tag. Temperature management of fresh products such as fresh foods or medicines should be performed thoroughly in its distribution processes. It requires high speed of read processing because a lot of time is taken when read a sensed data. And Cold Chain management system requires correct data which is read from the sensor tag. So, speedy transmission of a correct data is very important for managing the fresh products.

A lot of product management systems use a semi-passive sensor tag data for managing the products. Semi-passive sensor tag provides a function to store sensor information[3]. The sensor information is stored to tag memory according to the logging time interval. Semi-passive sensor tag provides two reading modes for getting a sensor data. One is getting current sensor value from a tag. This method simply gets the sensor value which is returned by the sensor attached to the semi-passive sensor tag. Another reading mode is getting logged sensor value from a tag.

Products management system using sensor tag is shown in Figure 1. Sensor tag starts logging a sensor data in tag memory when product is packaged in the factory. Sensing value is continuously logged while products are transported by vehicle. The reader

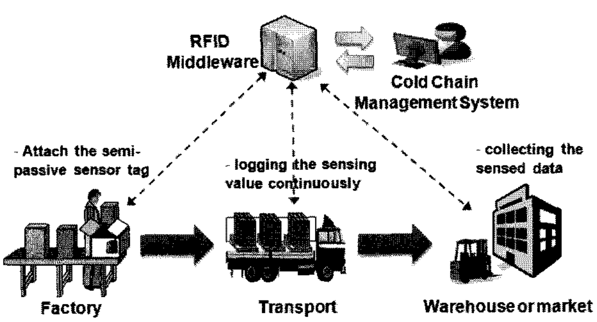


그림 1. 상품 관리 시스템
Fig. 1. Products managements system.

read the sensed data when products are arrived at destination which is warehouse or market because products should be checked for management safely.

Ⅲ. Problem in reading stored sensor data

Sensor values which are acquired from tagged objects are used momentously in applied environment. A large amount of sensor data is read at a time usually. Therefore data loss can be occurred during reading sensor data. Data loss is occurred because of the jamming or obstacle in the reader range. Both cases generate data loss while a sensor data is read from the tag by a reader. So, transmission of a tag data will be interrupted due to the reasons above.

The middleware cannot get complete data because of noisy environment as well as moving of the tag over reader range. If enter and leave of tag are repeated over reader range, data loss is occurred in the middle of result. And if the tag moves to other reader range, data loss could be occurred while tag is moved. At this time another reader read this tag again, so time may be wasted by reading a duplicated data.

RFID system using wireless communication cannot guarantee reliable transmission of the sensed data always. A Jamming is often occurred by the radio wave of many electronic products interfere with RF signals of readers. Data transmission may be affected by obstacles where data loss rate depends on the RF signal strength in target reader range.

In this paper, we present the problems which are generated by data loss. Memory Full may be occurred due to failure of reading the sensor data repeatedly. The sensing data does not be deleted from the tag memory until application receives the whole sensed data. If sensing data is deleted before receiving whole sensing data completely, product management system cannot manage the fresh product safely. Memory Full causes loss of the sensor tag data which is sensed currently or previously. Figure 2 shows Memory Full caused by continuous readout

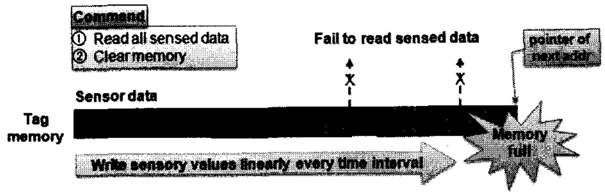


그림 2. 읽기 실패로 인한 Memory Full 발생
Fig. 2. Memory Full occurrence by read failure.

표 1. 환경에 따른 데이터 전송 결과
Table 1. Data Transmission results according to the environment.

<Use 16bytes packet size>

	Data loss rate	Elapsed time
Ideal environment	0% (0/4096bytes)	11sec
Noisy environment	96% (3920/4096bytes)	374sec

failure. It is generated when a reader incompletely read the sensed data continuously, because sensory values are written on semi-passive sensor tag memory continuously every specific interval.

Table 1 shows that data loss rate and elapsed time is different according to the environment status. If data loss is occurred while reading sensed data, it spends much time until return the result data as well as application cannot receive the correct result without data loss. So the reader read again to get a lost data. It spends much time because of big capacity of the sensed data. Most sensor tag has a lot of sensor data because a lot of sensor data are generated from factory to retailer. RFID middleware must read again the sensed data when data loss is occurred. It has an effect upon not only processing speed but also management of fresh products.

It is possible to consider air protocol to handle this problem. A lot of researches which is anti collision technique between readers and tags are existed currently. However, the problem of data loss is not solved completely by anti collision technique, and we cannot find method for solving problem of mentioned above. So RFID middleware requires reading method which is efficient reading for complete result data without data loss and speedy read for reducing requirement time.

Ⅳ. Transaction processing for efficient reading

1. Defining Transaction for Read Operation

Transaction processing consists of one or many physical operations (e.g. ReadMemory, Set PacketSize and so on) for getting a sensor data. The reading operation is processed when a Read command is send to the reader. Generally, only single reading process is executed in middleware when a Read command is send to read sensor data. However, it is difficult to guarantee complete reading using existing methods because it is hard to get correct sensed data through transmission of Read command at a time. Figure 3 shows that physical operation is executed by middleware for efficient reading when application command one logical operation to middleware in RFID system.

Results of the read commands are reported to the application either the complete result which is read successfully or the incomplete result which is failed to read when read operation is being processed in the RFID middleware. Besides, it takes long time for getting the sensed data when data loss is occurred. However, applications always want to receive complete results quickly to manage products. So we propose transaction processing for efficient reading of sensor data. It keeps executing Read operation one or many times according to the state of reading result until correct results are received. Therefore, several physical operations are executed for efficient read processing when application commands one logical operation to RFID middleware.

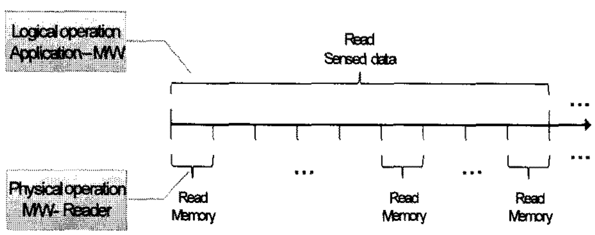


그림 3. 물리적 연산에 의한 트랜잭션 처리
Fig. 3. Transaction Processing by physical operation.

2. Transaction Processing using Incremental Reading

The data transmission is affected by internal condition of reader range. Table 2 shows that the data transmission rate and error rate depend on the packet size. The transmission rate and error rate decreases as the packet size goes smaller. Larger packet size makes higher transmission rate and error rate. Small packet size is favorable in noisy environment, and large packet size is favorable in general environment according to the error rate of each packet size. So we propose to use dynamic packet size technique which controls the packet size according to the environment condition.

IR (Incremental Reading) technique is that reads sensor data one by one according to the packet size from the semi-passive sensor tag memory. *IR* is transaction processing which use dynamic packet size technique for getting the sensor data and speeds up of read processing. Dynamic packet size technique is used in a noisy environment for minimizing data loss and to speed up the read processing. However, if reader read tag data using fixed packet size, it cannot control the transmission rate and error rate. So, we need to find suitable packet size for the target environment before reading the tag data and then read the sensed data from semi-passive sensor tag using found packet size.

The flow of Read operation in *IR* is illustrated in Figure 4. RFID Middleware sends a Read command to reader to read the sensed data by default packet size, and then middleware verify the result data. Controlling the packet size is in next step. If data loss is occurred continuously, packet size is decreased by this mechanism. Conversely, if data loss is not occurred repeatedly, packet size is increased by RFID

표 2. 패킷 크기 변화의 효과
Table 2. Effects of packet size variation.

Packet size	Transmission rate	Error rate
Big	High	High
Small	Small	Small

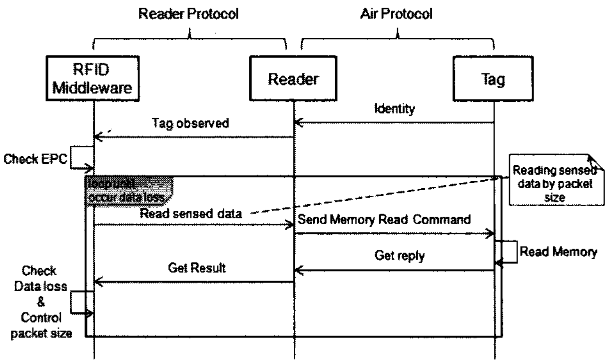


그림 4. IR 기법에서의 읽기 연산
Fig. 4. Read Operation in IR.

middleware. Read commands are sent to reader repeatedly for getting of complete result until reading point arrival at end point of the sensed data.

A control flow chart of *IR* technique is shown in Figure 5 First of all middleware choice the default packet size which is max or min in the Step 1, then Step 2 that read a data only by packet size is invoked. If data loss is occurred in the Step 2, RFID middleware sends read command again to reader for getting of lost data. It reads sensed data using same packet size until fail to read n times continuously. The opposite case, middleware checks state of reading is finished or not. And then transaction processing is finished if current reading address point is equal to end point of a data, Step 3 is invoked in opposite case. In Step 3, read point is moved to next, then Step 1 is invoked again. In Step 4 and Step 5, packet size is controlled according to the state of reading. For easy of explanation, packet size is

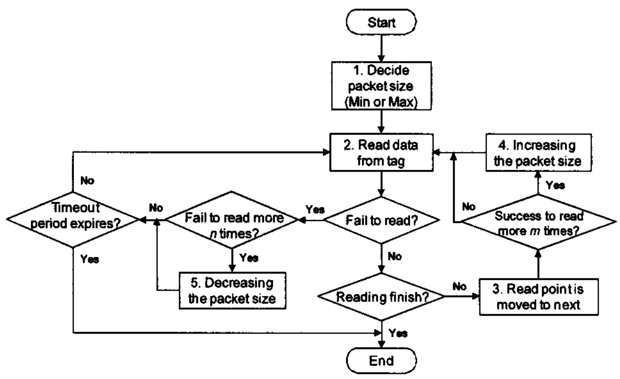


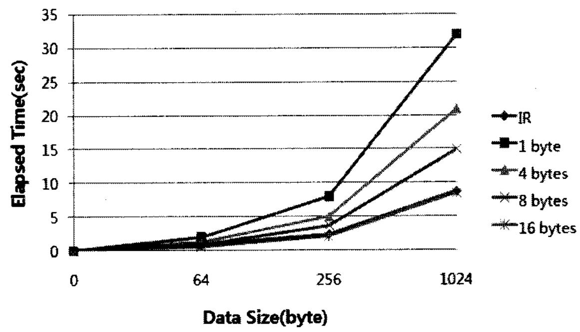
그림 5. IR 기법의 제어 흐름도
Fig. 5. Control flow of IR technique.

increased if data sensing is succeeded continuously, otherwise packet size is decreased. Transaction can be finished when timeout event is occurred or all sensed data is read completely.

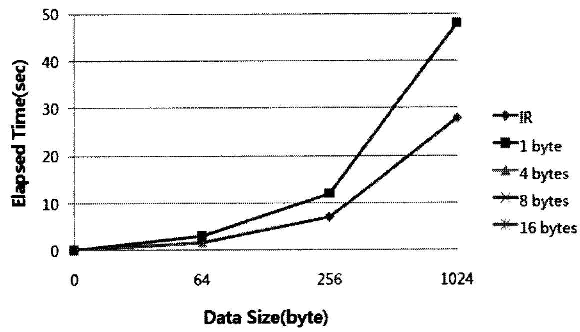
IR technique guarantees efficient reading in the noisy environment as well as the ideal environment. If we use *IR* method which uses dynamic packet size technique, middleware can cope with changeable status of reader range. It can process Read operation faster in noisy environment. However, Read operation requires much processing time comparatively because of reading sensed data one by one in ideal environment.

V. Performance Evaluation

We test proposed techniques, and confirm the guarantee of complete reading in noisy environment with obstacles. We use default packet size which is 1byte, 4bytes, 8bytes and 16 bytes in test. And it



(a) Test results in ideal environment



(b) Test results in noisy environment

그림 6. 제안한 기법의 성능평가 결과
Fig. 6. Performance evaluation of proposed technique.

read a sensed data of 1024 bytes from semi-passive sensor tag.

Result of reading a sensed data in ideal environment without jamming or obstacle is illustrated in Figure 6(a). In this graph, we compare proposed techniques with simple Read operation which is to use static packet size (i.e. 1, 2, 3 etc.). Each method guarantees complete reading in ideal environment. However, reading time varies according to the packet size. Using large packet size shows better performance than using small packet size in ideal environment.

Not each of the methods can guarantee efficient reading for getting of complete result data in noisy environment. And there is considerable performance gap among each method. Figure 6(b) shows that results of test using packet size of static or dynamic in noisy environment. General method which uses static packet size is not able to finish reading of the sensed data except only one packet size which is 1byte because it does not have re-read method for lost data. On the other hand, *IR* techniques guaranteed efficient reading for getting of complete result data and high speed of transaction processing in noisy environment. *IR* technique spends doubled time than the general method which uses static packet size of 1 byte. And general method may not finish reading semi-passive sensor tag data owing to timeout.

If middleware does not use the transaction processing like *IR* technique that we proposed, it is hard to find the packet size which is suitable for readout in specific environment, and it may make fail to read the sensed data although a tag is existed in the reader range. If middleware does not recover a lost data, the result is discarded with making Fail message to the application. We can get the complete result data which is sensing data of semi-passive sensor tag quickly by proposed *IR* technique.

Ⅵ. Related Work

IBM WebSphere RFID System[4] can be used to track goods in the supply chain, and manage business assets such as cars or computers and many other potential business applications. That system is implemented according to the ALE (Application Level Events)[5] standard in EPCglobal[6]. ALE standard specifies an interface through which application may obtain filtered and consolidated Electronic Product Code (EPC) data from a variety of sources.

RFID system may have a lot of readers, applications and queries. These factors raise increase in query processing time and network traffic. In previous software or research[4, 7~8] these issues are resolved to improve the performance of the RFID system. But we cannot find guarantee of complete reading of a tag memory data in existing RFID systems. These RFID systems transmit error messages to applications when data loss is occurred. And those systems do not process anything for getting the lost data.

So far RFID system cannot guarantee the efficient reading of the mass data from semi-passive sensor tag. ALE standard specifies a lot of functions for passive tag. However, we cannot find function of reading sensed data completely. If Cold Chain management system uses current RFID system, it cannot manage the fresh product safely. Because current RFID system do not guarantees the complete result of reading sensor data and high speed of read processing. Performance and functions of RFID system should be improved to satisfy with requirements of environment where sensor tag is used.

Ⅶ. Conclusion

It is important to get correct sensed data in logistics environment like Cold Chain. In this paper, we analyze the cases that relates to data loss. Data loss is occurred by jamming and obstacle in noisy environment. And we find common problems of each case and propose transaction processing method as a solution. We guarantee efficient reading through transaction processing when reader reads sensor data from semi-passive sensor tag using proposed *IR* technique.

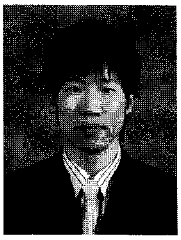
The technique that manages quality of products using sensor data has been used widely. In these cases, it is important to read the sensor data completely. So we propose the transaction processing to get the correct sensor data quickly, it can perfectly manage the fresh products. If RFID system uses transaction processing for efficient reading in Cold Chain environment, the reliability of reading a sensor data can be guaranteed. And proposed transaction processing shows improved performance that provide dependability of reading operation and high speed over reading the mass data. We confirm that improved performance using *IR* technique through performance evaluation in noisy environment.

The main contribution of our work is that we have proposed *IR* transaction processing for efficient reading of semi-passive sensor tag anywhere. Future research includes processing of the sensor data which is semi-passive sensor tag as well as active sensor tag, and extension of Middleware for Sensor Network.

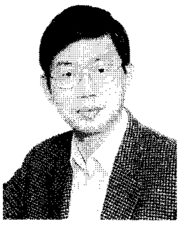
References

[1] EPCglobal Inc., The Tag Class Definitions, Version 1.0, http://www.epcglobalinc.org/standards/TagClassDefinitions_1_0-whitepaper-20071101.pdf
[2] Roy Want, "An Introduction to RFID Technology", IEEE Pervasive Computing, vol. 5, no.1, Jan-Mar 2006, pp. 25-33
[3] Reiner Jedermann, Luis Ruiz-Garcia, Walter Lang, "Spatial temperature profiling by semi-passive RFID loggers for perishable food transportation", Computers and Electronics in Agriculture, Elsevier B.V, Aug 2008, pp. 1-10.
[4] IBM., http://publib.boulder.ibm.com/infocenter/pvc-sensa/v1r1m0/index.jsp
[5] EPCglobal Inc., The Application Level Event (ALE) Specification, Version 1.1, http://www.epcglobalinc.org/stan-dards/ale/ale_1_1-standard-core-20080227.pdf
[6] EPCglobal Inc., http://www.epcglobalinc.org
[7] Joseph E. Hoag, Craig W. Thompson, "Architecting RFID Middleware", IEEE Internet Computing, vol. 10, no. 5, Sep-Oct 2006, pp. 88-92.
[8] Christian Floerkemeier, Matthias Lampe, "RFID middleware design: addressing appli-cation requirements and RFID constraints", ACM International Conference Proceeding Series, ACM, Grenoble, France, Oct 2005, pp. 219-224.
[9] Chen-Wei Lee, Chu-Sing Yang, and Yih-Ching Su, "Adaptive UEP and Packet Size Assignment for Scalable Video Transmission over Burst-Error Channels", EURASIP Journal on Applied Signal Processing, Hindawi Publishing Corp, Los Angeles, America, Oct 2006, pp. 1-9.
[10] D. Gras, "RFID based monitoring of the coldchain", Proceedings of the 2nd international Workshop Cold Chain Management University Bonn, May 2006, pp. 81-82.

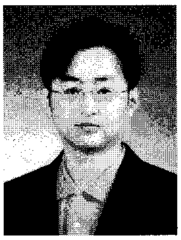
저 자 소 개



김 수 한(학생회원)
2007년 부산외국어대학교 컴퓨터공학과 학사 졸업.
2007년 9월~현재 부산대학교 대학원 컴퓨터공학과 석사과정.
<주관심분야 : RFID 미들웨어, Sensor tag 데이터 처리, RFID 태그 데이터의 효율적인 읽기 방법>



홍 봉 희(정회원)
1982년 서울대학교 전자계산기공학과 학사 졸업.
1984년 서울대학교 대학원 전자계산기공학과 석사 졸업.
1988년 서울대학교 대학원 전자계산기공학과 박사 졸업.
1987년~현재 부산대학교 컴퓨터공학과 교수.
<주관심분야 : RFID 미들웨어 데이터베이스, 실시간 위치정보 시스템, 유비쿼터스 미들웨어>



류 우 석(학생회원)
1997년 부산대학교 컴퓨터공학과 학사 졸업.
1999년 부산대학교 대학원 컴퓨터공학과 석사 졸업.
2002년 9월~현재 부산대학교 대학원 컴퓨터공학과 박사과정.
<주관심분야 : RFID 미들웨어, ALE, 태그 데이터 모델, RFID 트랜잭션, RTLS 위치 측위>